© 2025 IJRTI | Volume 10, Issue 1 January 2025 | ISSN: 2456-3315

Survey On Voice Command Based Data Retrieval System

[1] Gowri M Nair, [2]Shabna Shajahan ,[3]Clair Mary Mathew ,[4]Maria Joseph,[5]Rakhi Ramachandran Nair

[1,2,3,4]Student,[5]Assistant Professor

[1,2,3,4,5]Department of Computer Science and Engineering

[1,2,3,4,5]College Of Engineering Kidangoor,Kottayam, Kerala, India

[1]gowrimnair19@gmail.com , [2]shabnashajahan003@gmail.com , [3]clairmarymathew@gmail.com , [4]mariajoseph2310@gmail.com , [5]rakhinairtvla@gmail.com

ABSTRACT:The rapid advancements in natural language processing (NLP), speech recog- nition, and database management have led to the creation of innovative systems that improve how humans interact with computers. This survey combines information from ten research papers to provide a detailed overview of modern techniques in areas like voice-based interfaces, converting natural language into SQL queries, and deep learning methods for speech and text processing. Key topics include creating systems to turn nat- ural language into structured database queries, improving the accuracy of speech-to-text technology, and using advanced machine learning models such as Recurrent Neural Net- works (RNNs), Transformers, and Hidden Markov Models (HMMs). These developments aim to make human-computer interactions smoother and more effective. The surveyed pa- pers also points to specialized applications, such as smart home systems, big data models, and information retrieval from unstructured speech datasets. While noticeable progress has been made in areas like sequence modeling, feature extraction, and query optimiza- tion, the review identifies challenges related to scalability, noise handling, and language diversity. The production of these findings provides a roadmap for future research, the need for robust, adaptable, and user concentrated systems. This survey serves as a base reference for researchers seeking to address the technological gaps and advance innovations in natural language interfaces, speech recognition, and intelligent data retrieval systems.

INDEX TERMS: NLP, Speech Recognition, SQL Query Generation, RNNs, Transformer Models.

I.INTRODUCTION

In recent years, the rapid advancements in Artificial Intelligence (AI) and Natural Language Processing (NLP) have revolutionized human-computer interactions, introducing systems that prioritize simplicity, accessibility, and intuitive user experience. Among these, voice-command-based data retrieval systems have emerged as a vital innovation, enabling users to interact with databases using natural voice com- mands instead of conventional text-based query languages like SQL. These systems aim to simplify data retrieval processes, especially for non-technical users who lack proficiency in query languages, as well as for individuals with physical disabilities or those operating in hands-free environments such as industrial settings or while multitasking. By leveraging speech recognition and NLP, such systems in- terpret spoken inputs, convert them into text, and generate structured SQL queries to fetch and display the required data. This approach makes data retrieval more inclusive and user-friendly, addressing the limitations of traditional methods.

The concept of voice-based query systems has been explored extensively in existing research. Studies on voice-based natural language query processing have demonstrated how spoken queries can be pro- cessed into SQL commands through a combination of tokenization, syntactic analysis, semantic map- ping, and execution. However, these systems are constrained by ambiguities in natural language and a reliance on predefined dictionaries, which reduce their effectiveness for complex or diverse phrasing.

Another study focusing on data models for big data examines the scalability and flexibility of NoSQL frameworks in managing large-scale, high-velocity datasets. Although these frameworks address issues of scalability and heterogeneity, their lack of standardized schemas and consistency presents significant challenges for integration with voice-driven database systems.

Speech recognition technologies have played a crucial role in enhancing voice-based interfaces. Tech- niques such as Linear Predictive Coding (LPC), Hidden Markov Models (HMM), and Artificial Neural Networks (ANN) have been utilized to improve the accuracy of recognizing spoken commands. Despite these advancements, challenges such as handling different accents, background noise, and real-time processing remain persistent. Similarly, systems integrating NLP tools for query parsing and SQL syn- tax generation have shown promise in enabling non-technical users to interact with databases. These systems tokenize and analyze spoken queries to construct SQL commands but are often limited to basic query types, lacking support for advanced features like joins or context-aware queries.

Recurrent Neural Networks (RNNs) with Long Short-Term Memory (LSTM) models have intro- duced innovative methods for audio-to-text conversion, enabling smoother interactions in chatbot sys- tems. However, their effectiveness is hindered by scalability issues, limited training datasets, and high computational demands. Advanced frameworks like SQL-Net, Grammar SQL, and RAT-SQL employ schema linking, syntax trees, and attention mechanisms to enhance the accuracy of SQL query gener- ation. These frameworks demonstrate improved performance in handling complex queries but struggle with cross-domain applications, intricate schemas, and the ambiguity of natural language inputs.

Fuzzy retrieval techniques have been introduced to address challenges in processing large, unstruc- tured speech datasets, particularly for applications like information retrieval from news broadcasts. By utilizing phoneme-based matching and dynamic programming techniques, these methods achieve im- proved accuracy but require substantial preprocessing and computational resources, making them less viable for real-time scenarios. In parallel, smart home systems integrating IoT, fog,and cloud computing with NLP-driven voice interfaces demonstrate practical applications of speech recognition. The major limitation systems are adapting to languages with insufficient speech-to-text (STT) and text-to-speech (TTS)support, as well as tonal languages. For speech recognition, emotional analysis, speaker iden- tification, and speech quality enhancement Deep neural networks (DNNs) have been widely reviewed . The use of DNNs in automatic speech recognition systems highlights their reliance on specific feature extraction methods such as Mel-frequency cepstrum coefficients(MFCCs) and Word Error Rate (WER) evaluations.

The application is often constrained by a lack of language diversity, limited exploration of hybrid models, and challenges innoisy environments. Research on deep learning-driven natural language to SQL conversion further explores encoder-decoder frameworks, schema linking, and attention mechanisms to generate contextually accurate SQL queries and these approaches still face scalability challenges and struggle with capturing the complexities of nested queriesand intricate database structures..

Even though there has been a lot of progress in voice-command systems, there are still some big problems to fix. These include understanding what users really mean, making the system work well in different situations, and improving how efficiently the system runs. This paper suggests a new system that uses voice commands to help people to find information in databases. It combines speech recog- nition, language processing, and database queries to make the system better. By solving some of the problems found in previous research, the new system can make it easier to use, reduce mistakes, and help users find information faster and more easily. The study also points out areas that need more work, like creating systems that understand context, supporting more languages, and building smarter models. This will help improve voice technology in the future.

II. LITERATURE SURVEY

[1] The Voice-Based Natural Language Query Processing system is designed to allow users to interact with databases using spoken language without the need for SQL knowledge. The methodology consists of turning spoken queries into text, tokenizing and analyzing each term, and mapping to database terms in the dictionary. Syntactic and semantic analysis is then performed to create SQL queries, which are then executed on a relational database and results are displayed in a tabular format.

The system also learns from user questions to become more accurate over time, though it still struggles with complex or unusual phrasing. Although the approach is user-friendly and appropriate for handling complex queries, it is constrained by ambiguities in natural language and reliance on predefined dictionary mappings, which may make it less accurate for diverse or complex phrasing.

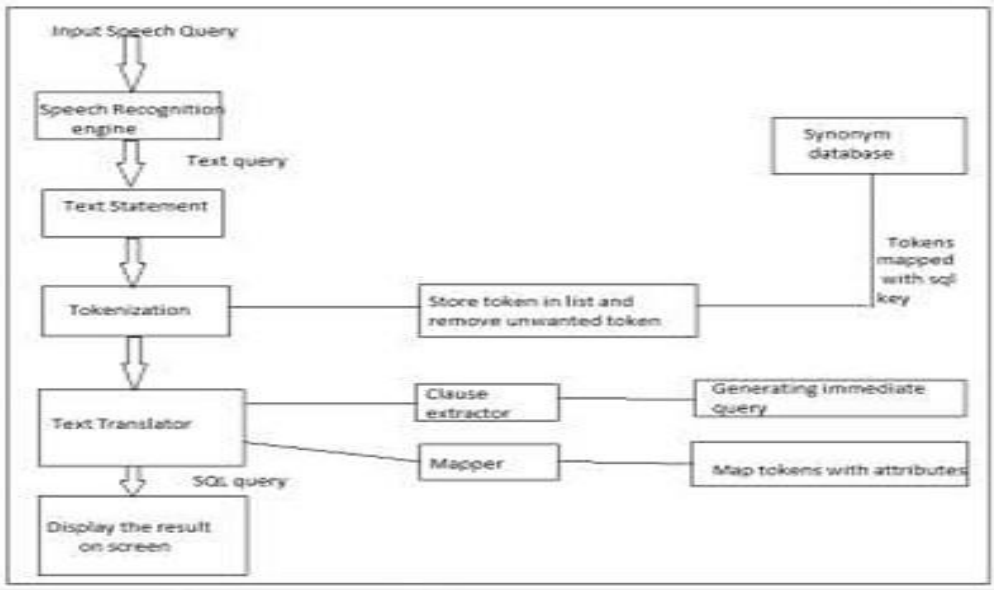

Figure 1: NLP Architecture [1]

Fig. 1 illustrates the architecture of the Natural Language Processing (NLP) system used in the voice-based query processing system. It is shown that the user begins by providing a speech input, which is then through speech recognition it is converted into text which is then tokenized into meaningful components, followed by lexical analysis to identify the role of each token. To interpret the structure and meaning of the query this system performs syntactic and semantic analysis. From this analysis, the system generates an SQL query that is executed on the database, and the results are returned to the user. This architecture helps bridge the gap between natural language and structured database queries, allowing for hands free database interactions. However, the system's reliance on predefined mappings and the complexities of natural language often limit its performance with more complex queries. [2] In order to meet big data concerns, this study thoroughly examines a number of data models, with an emphasis on models that manage high-volume, high-velocity, and high variety data.The study contrasts more modern NoSQL models, which provide improved scalability, flexibility, and horizontal distribution for massive data processing, with more conventional relational models. The major drawback includes complexity and heterogeneity of NoSQL models, which lack a common schema and frequently sacrifice consistency in favor of scalability and flexibility. Because of this it may be difficult to successfully integrate several data sources into a single framework .

[3] The main objective of the many voice recognition and information retrieval systems is to improve human-computer interaction. It is mostly helpful for individuals with disabilities and aims to create a speech-based interface for retrieving information from digital libraries. The various technologies like Linear Predictive Coding (LPC) for converting speech to text, Hidden Markov Models (HMM) for recognizing words, and Artificial Neural Networks (ANN) helps to improve the accuracy of recognizing spoken commands. The initiative intends to improve the accessibility and usability of information access by combining various technologies. Various technologies used in the system methodology includes LPC it is used for feature extraction, turning spoken words into data for analysis.

Then, an ANN model which is trained to identify keywords from the speech input. It uses a backpropagation algorithm to refine its accuracy. When pronunciation or speaker accents vary the HMM which is applied to create statistical models that help recognizing isolated words. Together, these methods allow the system to understand user commands and retrieve relevant information from a structured digital library database managed by SQL.Various limitations of the system that make speech recognition less accurate are handling different accents,speech rates,and background noise. The ANN model also needs a large amount of training data and time, which makes it slower in real-time applications. Additionally, while the HMM approach works well for isolated words, it struggles with continuous speech and unstructured input. The system also relies on a quiet environment, limiting its effectiveness in noisy settings. These limitations indicate that more improvements are necessary to make the system broadly usable.

Fig. 2 shows the project model for speech user interface. It is the architecture of a speech-based interface which is used for information retrieval from digital libraries. The process begins with accepting speech input, which is converted into data using Linear Predictive Coding (LPC) for feature extraction. An Artificial Neural Network (ANN) evaluates the resulting data and uses a backpropagation technique to extract important keywords from the speech. For recognizing words we use Hidden Markov Models (HMMs), particularly for addressing variations in pronunciation and accents. The system aims to improve accuracy in recognizing spoken commands, although it faces challenges with continuous speech and unstructured input. This model is designed to provide a more intuitive interface for information retrieval, especially for users with disabilities.

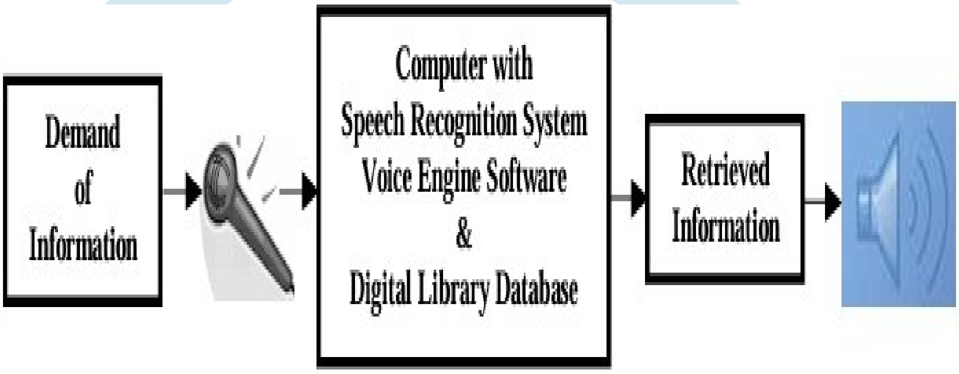

Figure 2: project model for speech user interface[3]

[4] By enabling voice-based natural language queries the system designed to simplify database interac- tions for non technical user . It also addresses the issue that laypeople often lack the technical skills to interact with databases using SQL, which can limit their access to needed information. The proposed system allows users to input queries through natural language speech then converted into SQL com- mands to retrieve data. To transform spoken language into structured database queries it uses NLP tools tokenization ,part of speech taggingand keyword extraction. It works by input the voice which is converted to text, followed by the tokenization and tagging of the natural language query (NLQ).

To parse and interpret the query it uses lexicon set containing SQL keywords and database attribute map. Based on identified SQL commands keywords are extracted and mapped to database elements (tables, columns). Finally, the system generates the SQL syntax tree to execute the query. Key NLP components, such as Stanford NLP tools, help structure the query formation, ensuring accurate repre- sentation of user requests.The major limitations of this system is currently restricted to specific types of SQL queries (e.g., select, where, group by) and lacks support for advanced features like joins and inner clauses and also it does not support follow-up or context-aware queries.

[5] Using Recurrent Neural Networks (RNNs)that helps the system to convert audio to text within a chatbot to create smoother, more responsive interactions.The system encodes audio inputs and de- codes them into text, allowing the chatbot to generate text-based responses using sequence-to-sequence RNN models. By identifying gender the model helps to make chatbot interactions more natural and even adapts to the user for more personalized responses. It is built with machine learning libraries such as Keras and PyTorch , emphasizing the innovative use of RNN-based encoding and decoding in audio-to-text conversion.For handling sequential data the system works mainly centered on RNN sequence-to-sequence models with Long Short-Term Memory (LSTM) . The process mainly contain two steps,that is encoding the audio signal into a dataset. Then decoding audio into text responses. By testing the system with libraries, the authors compared factors like training loops and model accuracy. For creating a user-friendly interface for deployment on platforms like Telegram it uses MySQL for data storage, and Bootstrap 4 was used for the frontend . The limitations of the system includes the relatively small training dataset restricts the model’s ability to perform well with more extensive or diverse data, which may reduce translation quality and generalization. Second, the performance of the model with Keras and PyTorch depends heavily on the available computational resources, suggesting scalability issues without substantial hardware support. Third, while Keras was chosen for its ease of use and lower resource requirements, this choice may limit optimization possibilities that more complex frameworks like TensorFlow could offer. Finally, the system’s focus on legal datasets may make it less adaptable to other contexts, reducing

its general applicability beyond similar structured text environ- ments.

[6] There are presently many methods existing to convert natural language(NL) inputs into SQL queries for database management. In this research paper it provides an extensive analysis of such existing frameworks that convert natural language inputs into SQL queries. The study highlights the significance of Natural Language Interfaces to Databases (NLIDBs), which allow users without SQL knowledge to query databases using natural language, thereby making databases more accessible. To see how well these different systems work, the study uses the SPIDER dataset, which is famous for having really complex SQL queries that cover a lot of different topics. The paper explores several mod- els developed over recent years, examining each framework's architecture, accuracy, and application in handling cross-domain and complex SQL queries.

It checks out several frameworks like SQL-Net, Syntax SQL-Net, Grammar SQL, IR-Net, Edit SQL, and RAT-SQL. SQL-Net framework uses a sketch-based approach to simplify query prediction, achieving moderate accuracy but struggling with cross-domain queries. It is introduced to minimize the use of the reinforcement learning by avoiding seq2seq in cases where an order is of low significance. Syntax SQL-Net framework introduces syntax trees to handle complex queries, improving accuracy by approximately 15 percent over SQL-Net.

The next framework Grammar SQL incorporates grammar rules to create SQL queries, reducing errors by ensuring table and column consistency. Grammar SQL sticks to strict grammar rules to make sure the SQL is organized correctly, but that can make it less flexible when dealing with natural language that isn't so structured. It incorporates grammar rules to create SQL queries, thereby reducing errors by ensuring table and column consistency. Then there's IR-Net, which uses BERT embeddings and schema linking to get a better understanding of context. It works well, but it also needs a lot of processing power to function. Edit-SQL allows for making changes to queries based on context dependency and previous predictions. The bestperforming model is RAT-SQL, which uses BERT and special transformers that know about relationships between data. It's really effective, hitting an accuracy of 65.6 percent of computer resources. Even though there's been a lot of progress, there are still some challenges, especially with complex schemas and unclear terms. In the future, researchers might want to work on making predictions better and understanding context more deeply, maybe by using ontological directed graphs.

[7] "Information Retrieval Based on Speech Recognition Results" talks about a fuzzy retrieval method that deals with problems in transcribing spoken language, especially when it comes to big audio datasets like news broadcasts.It mainly describes the challenges of information retrieval in large, untagged speech datasets. Existing traditional methods are resource-intensive and often inaccurate due to recognition errors.

The overview of the retrieval system outlines the system's process, where speech is divided then processed with a language model, and analyzed using non-linear matching techniques to account for recognition errors. This method introduces two cool techniques to help make retrieval more accurate: first, there's a phoneme distance matrix that looks at the differences in sound between phonemes, and second, there's a confusion matrix that takes common phoneme mistakes into account.Regular retrieval methods don't really cut it when there are a lot of errors from speech-to-text conversions. So, they use dynamic programming (DP) to line up and match the recognized text with queries after turning the audio data into text with the Julius speech recognition system.

In a study with 30 people, researchers found that music preferences are totally connected to emotion, what you're doing, and your environment. For example, your music taste can change depending on whether you're happy, sad, working out or just chilling. The bracelets even picked up on heart rate, and while it mattered a little less, it still helped shape music choices. The researchers came up with a cool new system called the Multi-task Ubiquitous Music Recommendation model, or MUMR for short. This model takes all the information from the smart bracelets and uses it to predict what kind of music you'll like at any given moment. It's a better way to recommend songs because it doesn't just guess based on what you listened to before—it uses real-time data about how you're feeling and what's going on around you. The model improves music recommendation performance by incorporating bracelet-collected context as an input and mood prediction as an auxiliary task. This approach not only enhances the personalization of music recommendations but also shows how wearable technology can bridge the gap between digital systems and real-world experiences.

Experimental results show that this fuzzy retrieval method is about 6 percent more accurate than just using exact matches.While this method is efficient, it needs a lot of preprocessing to create the language models, and it requires a ton of computing power, so it might not be great for real-time ap- plications. Also, the method might not work as well with different languages or dialects, which means they need to do more testing with a variety of datasets to make it stronger.

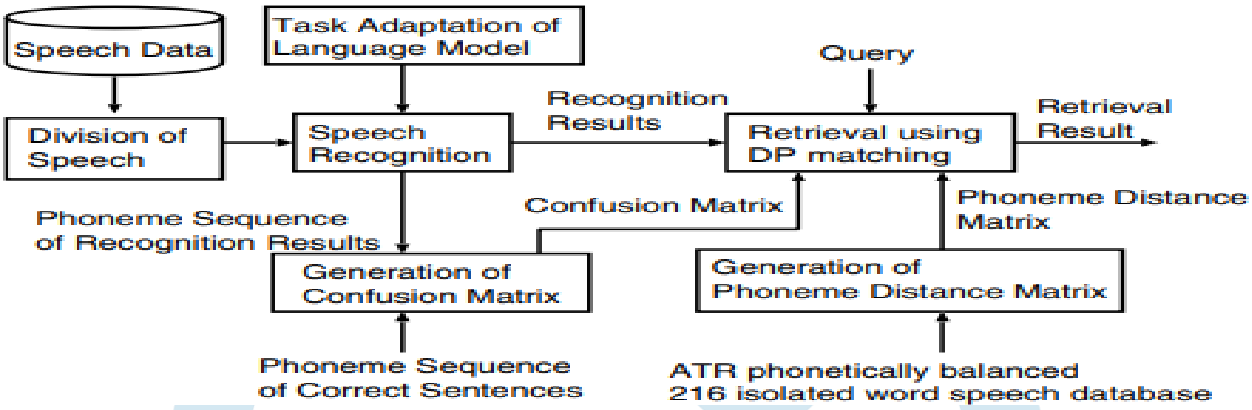

Figure 3: overview of retrieval system [7]

Fig. 3 illustrates the fuzzy retrieval method employed to address problems in speech recognition. It is particularly used when dealing with large, untagged audio datasets. The process starts by segmenting the speech data into smaller sections. A language model then processes these segments, and to account for recognition errors non-linear matching techniques, including phoneme distance and confusion ma- trices, are applied. To align and match the recognized text with predefined query patterns Dynamic programming (DP) is used which improves search accuracy. This enhances retrieval over traditional exact match methods but requires significant preprocessing and computational resources, limiting its use in real-time applications.

[8] Focusing a smart home system that mixes Internet ofThings (IoT), fog computing, and cloud computing with a voice user interface (VUI) that uses natural language processing (NLP), especially designed for languages that don't get a lot of support, like Bulgarian. This IoT-Fog-Cloud (IoTFC) setup brings together IoT sensors, NLP, and cloud services to handle voice commands both locally and online, which helps make the system faster, safer, and more flexible. It spread out across different locations, plus a fog-level integration server by the way it works includes using speech-to-text (STT) and text-to-speech(TTS) services. This means it doesn't just rely on processing everything in the cloud. Even though this system does a good job of understanding voice commands for specific tasks, it has some limitations. For example, it might not work well with languages that don't have STT or TTS services, and it could struggle with tonal languages.Overall, the study provides a practical way that control smart homes with your voice, filling some gaps that exist in commercial systems by supporting languages that aren't as popular. In areas related to speech, like automatic speech recognition,emotional speech recognition, iden- tifying speakers, and improving speech quality it discusses how deep neural networks (DNNs) are used. It reviews 174 studies from 2006 to2018 and breaks them down based on different factors like thetype of speech, the datasets used, languages, environments,methods for extracting features, evaluation metrics, and the types of DNN models. The findings also shows that there's a big focus on automatic speech recognition and that most studies use public datasets in English. To do this review the researchers fol- lowed a Systematic Literature Review (SLR), which included figuring out research questions, planning how to search for studies, picking which studies to include, and extracting the data. This process led to some useful insights into what's common in DNN-based speech recognition. They found that most studies rely heavily on Melfrequency cepstrum coefficients (MFCCs) for extracting features, use Word Error Rate (WER) as the main way to measure performance, and mostly stick to standalone DNN models. There's been less exploration of recurrent neural networks (RNNs) and hybrid models. Lack of language diversity, a focus on neutral environments instead of noisy or emotional ones, and the heavy reliance on MFCCs are some key limitations they pointed out . The researchers suggest that future studies could really benefit from using more diverse datasets, hybrid methods, and different ways to evaluate models to make them more robust and useful across various languages and situations.

[9]

[10] Deep learning driven natural languages text to sql query conversion explores various methodologies employed in deep learning-driven natural language to SQL query conversion, which is a critical task in natural language processing (NLP) and database management. Encoder-decoder framework is one of the most common approaches , which forms the basis of many models. In this framework, the encoder processes the input natural language query, converting it into a latent representation that captures the semantic meaning of the input. The translation of this representation into a structured SQL query is done by the decoder. This approach is well-suited for tasks that involve sequence-to-sequence learning, such as translating human-readable queries into formal query languages like SQL.

Schema linking is the key feature of this methodology , where the model maps natural language enti- ties (like table names, column names, and values) to the corresponding elements in the database schema. It also ensures that the SQL query is contextually correct and executable on the target database. Ad- ditionally, attention mechanisms are incorporated into many models to help them focus on the relevant parts of both the input query and the schema during the translation process. This allows the model to handle more complex queries with intricate relationships between tables and columns, which is crucial when dealing with SQL queries that involve joins, aggregations, or subqueries. Various deep learning architectures, including Convolutional Neural Networks (CNNs) for capturing local dependencies, Re- current Neural Networks (RNNs) for sequence modeling, Pointer Networks for handling schema-based selection, and Transformer Models for capturing long-range dependencies within the input text,the model evaluted in the paper use these techniques.

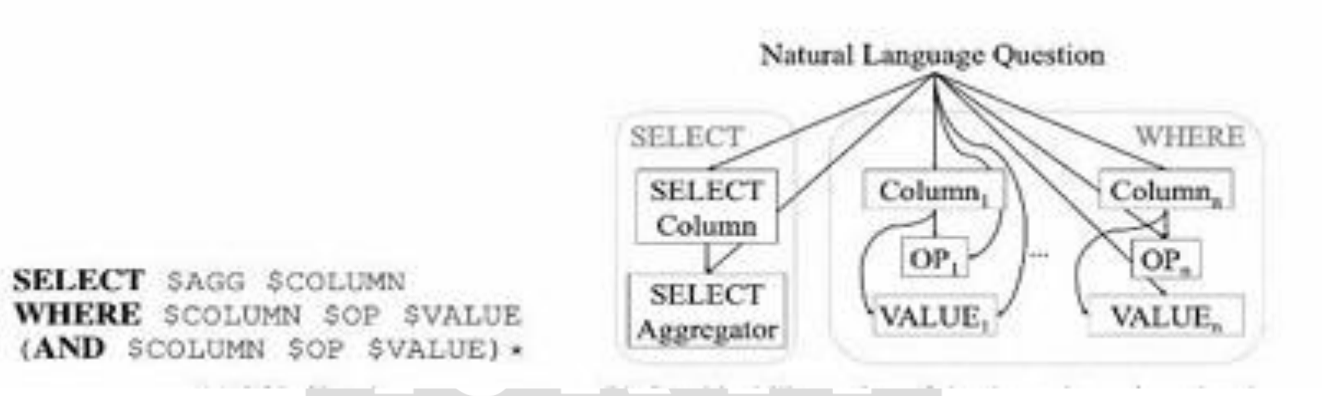

Figure 4: SQL sketch and graphical illustration of dependency in sketch [10]

Fig. 4 provides a graphical representation of how a natural language query is transformed into an SQL query. First, the input query in natural language is processed to identify entities such as table names, column names, and values. These entities are then linked to the corresponding elements in the database schema. A ”sketch” or template of the SQL query is generated, outlining its structure. The dependencies between different components of the query, such as relationships between tables and columns, are represented in the diagram. This visual representation highlights the importance of schema-aware models and attention mechanisms in generating accurate SQL queries, especially when handling complex SQL constructs like joins and subqueries.

The datasets used to train and test these models include well-known ones like Spider and WikiSQL. These datasets are commonly used to check how well the models perform. The training focuses on teaching the models to understand the connections in natural language queries like the purpose, table names, and column names and accurately translate them into SQL structures.One important idea in the paper is the need for schema-aware models. These models include information about the database schema, which helps them figure out how different tables and columns are related. This is crucial for creating SQL queries that are not only correct in structure but also make sense. However, SQL queries can be quite complicated, often involving things like nested queries, joins, and grouping. Even with advanced techniques like attention mechanisms, many models still find it hard to handle these tricky parts.Scalability is another major issue. Models that use complex systems like transformers need a lot of computing power to train. And while there has been progress in automating the process of turning natural language into SQL, these systems are still not widely used. The main reason is that it’s really hard to fully cover all the details of SQL syntax and semantics, which is necessary for generating queries that work efficiently.These problems show that more research is needed. Specifically, there’s a need to make models more adaptable, train them on more diverse datasets, and create better ways to handle complex SQL structures.

III. CONCLUSION

The Voice-to-Text SQL Query Project combines speech recognition, natural language processing, and database management into a tool . It makes retrieving data more easier, especially for people with dis- abilities, and improve productivity by enabling users simply speak their queries instead of typing them out. It ensures accuracy making it more reliable . Those who have without a technical background can quickly and comfortably query databases by using this simple interface.The project opening doors for even more innovative uses in the future by demonstrates exciting potential for voice-activated data tools across various fields. This system is paving the way for easier and friendlier database interactions for more people.

IV. REFERENCES

[1] Munde, Puja, Sayali Tambe, Afreen Shaikh, Pratiksha Sawant, and Deepa Mahajan. ”Voice based natural language query processing.” Int. Res. J. Eng. Technol 7 (2020).

[2] F. Mostajabi, A. A. Safaei and A. Sahafi, ”A Systematic Review of Data Models for the Big Data Problem,” in IEEE Access, vol. 9, pp. 128889-128904, 2021.

[3] Shrawankar, Urmila, and Anjali Mahajan. ”Speech user interface for information retrieval.” arXiv preprint arXiv:1305.1429 (2013).

[4] Wahi, Varun, Avadhoot Joshi, Rohan Patel, Asad Mujawar, and Nishant Tayde. ”Voice Con- trolled Database Analysis” .

[5] Basystiuk, Oleh, Natalya Shakhovska, Violetta Bilynska, Oleksij Syvokon, Oleksii Shamuratov, and Volodymyr Kuchkovskiy. ”The Developing of the System for Automatic Audio to Text Conversion.”In ITAS, pp. 1-8. 2021.

[6] Baig, Muhammad Shahzaib, Azhar Imran, Aman Ullah Yasin, Abdul Haleem Butt, and Muhammad Imran Khan. ”Natural language to sql queries: A review.” International Journal of Innovations in Science Technology 4 (2022): 147-162.

[7] Watanabe, Masatoshi, and Masahide Sugiyama. ”Information retrieval based on speech recogni- tion results.” Threshold 800, no. 1000 (2002): 1200.

[8] Iliev, Yuliy, and Galina Ilieva. ”A framework for smart home system with voice control using NLP methods.” Electronics 12, no. 1 (2022): 116.

[9] Nassif, Ali Bou, Ismail Shahin, Imtinan Attili, Mohammad Azzeh, and Khaled Shaalan. ”Speech recognition using deep neural networks: A systematic review.” IEEE access 7 (2019): 19143-19165.

[10]Kumar, Ayush, Parth Nagarkar, Prabhav Nalhe, and Sanjeev Vijayakumar. ”Deep learning driven natural languages text to SQL query conversion: a survey.” arXiv preprint arXiv:2208.04415 (2022).